Listing of Claims

1. (Previously Presented) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

store a portion of a first Reed-Solomon code word, the portion being less than the entire first code word;

store a portion of a second Reed-Solomon code word, the portion being less than the entire second code word; and

while storing the portion of the second code word, decoding the portion of the first code word and no other portion of the first code word.

2. (Previously Presented) The computing system of claim 1 wherein:

storing the portions of the first and second Reed-Solomon code words comprises storing each of the portions in a respective time having a duration *t*; and

decoding the portion of the first code word comprises decoding the portion in the duration *t*.

Claims 3 -4 Canceled.

5. (Previously Presented) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

receive *m* portions of a first Reed-Solomon code word, *m* being greater than one;

receive *m* portions of a second Reed-Solomon code word after receiving the first code word;

while receiving a first portion of the second code word, decoding a first portion of the first code word and no other portion of the first code word; and

while receiving a second portion of the second code word, decoding the second portion of the first code word and no other portion of the first code word.

6. (Previously Presented) The computing system of claim 5 wherein the processor circuit is further operable to:

receive the first and second portions of the second code word during respective first and second time periods each having a same duration; and

decode the first and second portions of the first code word during the first and second time periods respectively.

7. (Previously Presented) The computing system of claim 5 wherein the processor circuit is further operable to:

receive *m* portions of a third Reed-Solomon code word;

while receiving a first portion of the third code word, decoding a first portion of the second code word and no other portion of the second code word; and

while receiving a second portion of the third code word, decoding the second portion of the second code word and no other portion of the second code word.

8. (Previously Presented) The computing system of claim 5 wherein:

the first and second Reed-Solomon code words each comprise *n* symbols; and

*n/m* equals an integer that is greater than one.

9. (Previously Presented) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

receive *m* portions of each of five Reed-Solomon code words during respective time periods each having a duration, each code word comprising *n* symbols such that *n/m* equals an integer that is greater than one;

while receiving a first portion of the second code word, decoding a first portion of the first code word according to a first algorithm and decoding no other portion of the first code word;

while receiving a first portion of the third code word, decoding a first portion of the second code word according to the first algorithm, decoding the first portion of the first code word according to a second algorithm, and decoding no other portions of the first and second code words;

while receiving a first portion of the fourth code word, decoding a first portion of the third code word according to the first algorithm, decoding the first portion of the second code word according to the second algorithm, decoding the first portion of the first code word according to a third algorithm, and decoding no other portions of the first, second, and third code words; and

while receiving a first portion of the fifth code word, decoding a first portion of the fourth code word according to the first algorithm, decoding the first portion of the third code word according to the second algorithm, decoding the first portion of the second code word according to the third algorithm, decoding the first portion of the first code word according to a fourth algorithm, and decoding no other portions of the first, second, third, and fourth code words.

10. (Previously Presented) A computing system for decoding a Reed-Solomon-encoded string of data, the computing system comprising a processor circuit operable to:

sequentially receive *m* portions of the Reed-Solomon-encoded string of data in *T* seconds, the string of data including *n* symbols, *n/m* equaling a first integer that is greater than one;

sequentially calculate *m* respective partial syndromes for the *m* portions of the string, the processor circuit operable to calculate each of the *m* partial syndromes in *T/m* seconds;

from the *m* partial syndromes, sequentially calculate the coefficients of *m* respective sets of error locator polynomials, the processor circuit operable to calculate each set of coefficients in *T/m* seconds;

sequentially determine *m* respective sets of roots for the sets of the error locator polynomials, the processor circuit operable to determine each set of roots in *T/m* seconds;

for each of the *m* sets of roots, sequentially determine the magnitude of a respective error in *T/m* seconds; and

sequentially correct each of the errors in *T/m* seconds.

11. (Previously Presented) The computing system of claim 10 wherein:

said *n* symbols each comprise b bits;

*k* of said n symbols comprise data symbols;

*n-k* of said symbols comprise parity symbols;

*(n-k)/m* equals a second integer; and

*k/m* equals a third integer.

Claim 12 Canceled.

13. (Original) The computing system of claim 10 wherein the processor circuit executes a Berlekamp-Massey Algorithm to calculate the coefficients of the error locator polynomial.

14. (Original) The computing system of claim 10 wherein the processor circuit executes a Chien Search to determine the roots of the error locator polynomial.

15. (Original) The computing system of claim 10 wherein the processor circuit executes a Forney Algorithm to determine the magnitude of the errors in the received digital code word.

Claims 16 - 18 Canceled.

19. (Previously Presented) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps of:

receiving a portion of a first Reed-Solomon code word, the portion being less than the whole first code word;

receiving a portion of a second Reed-Solomon code word, the portion being less than the whole second code word; and

while receiving the portion of the second code word, decoding the portion of the first code word and no other portion of the first code word.

20. (Previously Presented) The method of claim 19 wherein the portion of the first Reed-Solomon code word is the same length as the portion of the second Reed-Solomon code word.

21. (Previously Presented) The method of claim 19 wherein:

receiving the portion of the second code word comprises receiving the portion of the second code word in a time *T*; and

decoding the portion of the first code word comprises decoding the portion of the first code word in the time *T*.

22. (Previously Presented) The method of claim 19 wherein:

the first and second code words each comprise *m* portions of equal length.

23. (Previously Presented) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps of:

receiving *m* portions of a first Reed-Solomon code word, *m* being greater than one;

receiving *m* portions of a second Reed-Solomon code word after receiving the first code word; and

while receiving a first portion of the second code word, decoding a first portion of the first code word and no other portion of the first code word; and

while receiving a second portion of the second code word, decoding the second portion of the first code word and no other portion of the first code word..

24. (Previously Presented) The method of claim 23 wherein:

receiving the *m* portions of the first and second code words comprises receiving each of the *m* portions during a respective period of time T; and

decoding each of the first and second portions of the first code word during the same respective periods that the first and second portions of the second code word are received.

Claim 25 Canceled.

26. (Previously Presented) The method of claim 23 wherein:

the first and second code words each comprise *n* equal-length symbols; and

*n/m* equals an integer greater than one.

27. (Previously Presented) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps of:

receiving *m* portions of each of five Reed-Solomon code words during respective time periods each having a duration, each code word comprising *n* symbols such that *n/m* equals an integer that is greater than one;

while receiving a first portion of the second code word, decoding a first portion of the first code word according to a first algorithm and decoding no other portion of the first code word;

while receiving a first portion of the third code word, decoding a first portion of the second code word according to the first algorithm, decoding the first portion of the first code word according to a second algorithm, and decoding no other portions of the first and second code words;

while receiving a first portion of the fourth code word, decoding a first portion of the third code word according to the first algorithm, decoding the first portion of the second code word according to the second algorithm, decoding the first portion of the first code word according to a third algorithm, and decoding no other portions of the first, second, and third code words; and

while receiving a first portion of the fifth code word, decoding a first portion of the fourth code word according to the first algorithm, decoding the first portion of the third code word according to the second algorithm, decoding the first portion of the second code word according to the third algorithm, decoding the first portion of the first code word according to a fourth algorithm, and decoding no other portions of the first, second, third, and fourth code words.

28. (Previously Presented) A method of operating a computing system with a Reed-Solomon decoding application comprising the steps:

sequentially receiving *m* portions of the Reed-Solomon-encoded string of data in *T* seconds, the string of data including *n* symbols, *n/m* equaling a first integer that is greater than one;

sequentially calculating *m* respective partial syndromes for the *m* portions of the string, the processor circuit operable to calculate each of the *m* partial syndromes in *T/m* seconds;

from the *m* partial syndromes, sequentially calculating the coefficients of *m* respective sets of error locator polynomials, the processor circuit operable to calculate each set of coefficients in *T/m* seconds;

sequentially determining *m* respective sets of roots for the sets of the error locator polynomials, the processor circuit operable to determine each set of roots in *T/m* seconds;

for each of the *m* sets of roots, sequentially determining the magnitude of a respective error in *T/m* seconds and correcting each of the errors in *T/m* seconds.

Claims 29 – 30 Canceled.

31. (Original) The method of claim 28 wherein the computing system executes a Berlekamp-Massey Algorithm to calculate the coefficients of the error locator polynomial.

32. (Original) The method of claim 28 wherein the processor circuit executes a Chien Search to determine the roots of the error locator polynomial.

33. (Original) The method of claim 28 wherein the processor circuit executes a Forney Algorithm to determine the magnitude of the errors in the received digital code word.

Claims 34 – 36 Canceled.

37. (Previously Presented) The method of claim 27 wherein the portions of the five Reed-Solomon code words each have a same size.